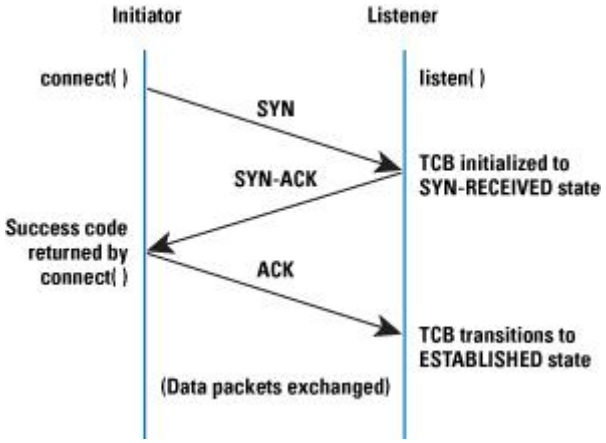
SYN Flood 攻击及防御方法

一、 为什么 Syn Flood 会造成危害

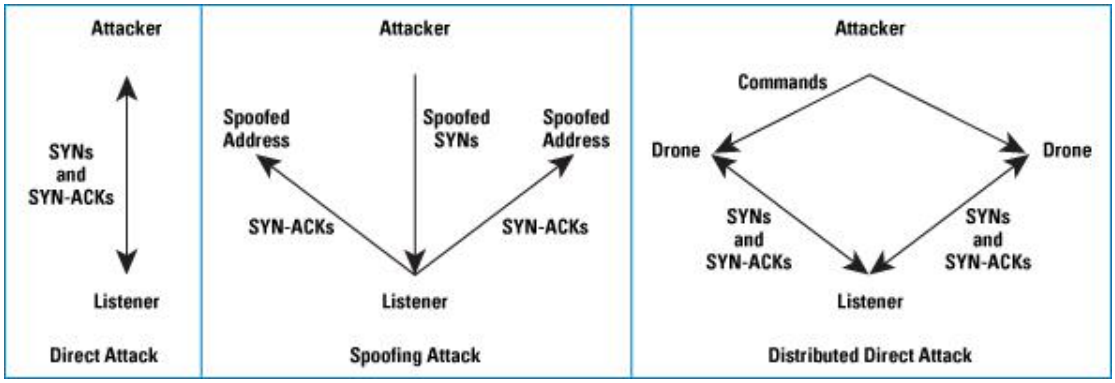
这要从计算机网络的 TCP/IP 协议栈的实现说起。当开放了一个 TCP 端口后，该端口就处于 Listening 状态，不停地监视发到该端口的 SYN 报文，一旦接收到 Client 发来的 SYN 报文，就需要为该请求分配一个 TCB（Transmission Control Block），通常一个 TCB 至少需要 280 个字节，在某些操作系统中 TCB 甚至需要 1300 个字节，并返回一个 SYN ACK 报文，立即转为 SYN-RECEIVED 即半开连接状态，而某些操作系统在 SOCKT 的实现上最多可开启 512 个半开连接（如 Linux2.4.20 内核）。这种过程如下图所示：



从以上过程可以看到，如果恶意的向某个服务器端口发送大量的 SYN 包，则可以使服务器打开大量的半开连接，分配 TCB，从而消耗大量的服务器资源，同时也使得正常的连接请求无法被相应。而攻击发起方的资源消耗相比较可忽略不计。

二、 如何防御 Syn Flood 攻击

我们先来看一下 Syn Flood 有哪些种类，如下图所示：



1．Direct Attack 攻击方使用固定的源地址发起攻击，这种方法对攻击方的消耗最小
2．Spoofing Attack 攻击方使用变化的源地址发起攻击，这种方法需要攻击方不停地修改源地址，实际上消耗也不大
3．Distributed Direct Attack 这种攻击主要是使用僵尸网络进行固定源地址的攻击

对于第一种攻击的防范可以使用比较简单的方法，即对 SYN 包进行监视，如果发现某个 IP 发起了较多的攻击报文，直接将这个 IP 列入黑名单即可。当然下述的方法也可以对其进行防范。对于源地址不停变化的攻击使用上述方法则不行，首先从某一个被伪装的 IP 过来的 Syn 报文可能不会太多，达不到被拒绝的阈值，其次从这个被伪装的 IP（真实的）的请求会被拒绝掉。因此必须使用其他的方法进行处理。

1.无效连接监视释放

这种方法不停监视系统的半开连接和不活动连接，当达到一定阈值时拆除这些连接，从而释放系统资源。这种方法对于所有的连接一视同仁，而且由于 SYN Flood 造成的半开连接数量很大，正常连接请求也被淹没在其中被这种方式误释放掉，因此这种方法属于入门级的 SYN Flood 方法。

2.延缓 TCB 分配方法

从前面 SYN Flood 原理可以看到，消耗服务器资源主要是因为当 SYN 数据报文一到达，系统立即分配 TCB，从而占用了资源。而 SYN Flood 由于很难建立起正常连接，因此，当正常连接建立起来后再分配 TCB 则可以有效地减轻服务器资源的消耗。常见的方法是使用 SYN Cache 和 SYN Cookie 技术。

SYN Cache 技术：

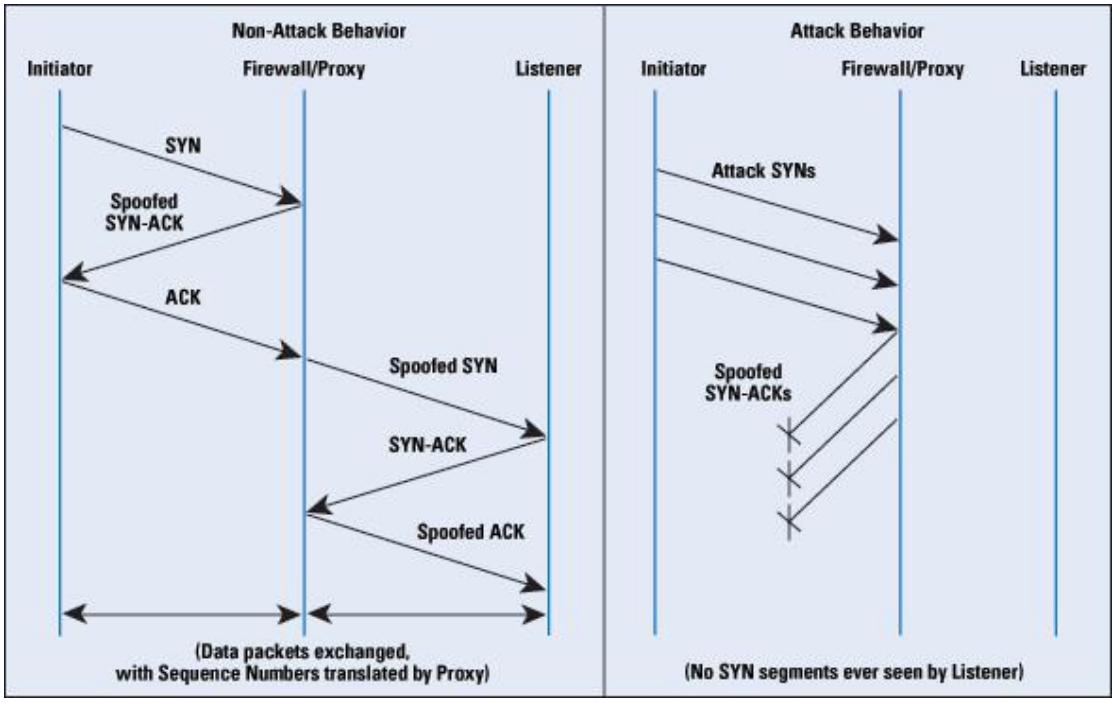
这种技术是在收到 SYN 数据报文时不急于去分配 TCB，而是先回应一个 SYN ACK 报文，并在一个专用 HASH 表（Cache）中保存这种半开连接信息，直到收到正确的回应 ACK 报文再分配 TCB。在 FreeBSD 系统中这种 Cache 每个半开连接只需使用 160 字节，远小于 TCB 所需的 736 个字节。在发送的 SYN ACK 中需要使用一个己方的 Sequence Number，这个数字不能被对方猜到，否则对于某些稍微智能一点的 SYN Flood 攻击软件来说，它们在发送 SYN 报文后会发送一个 ACK 报文，如果己方的 Sequence Number 被对方猜测到，则会被其建立起真正的连接。因此一般采用一些加密算法生成难于预测的 Sequence Number。

SYN Cookie 技术：

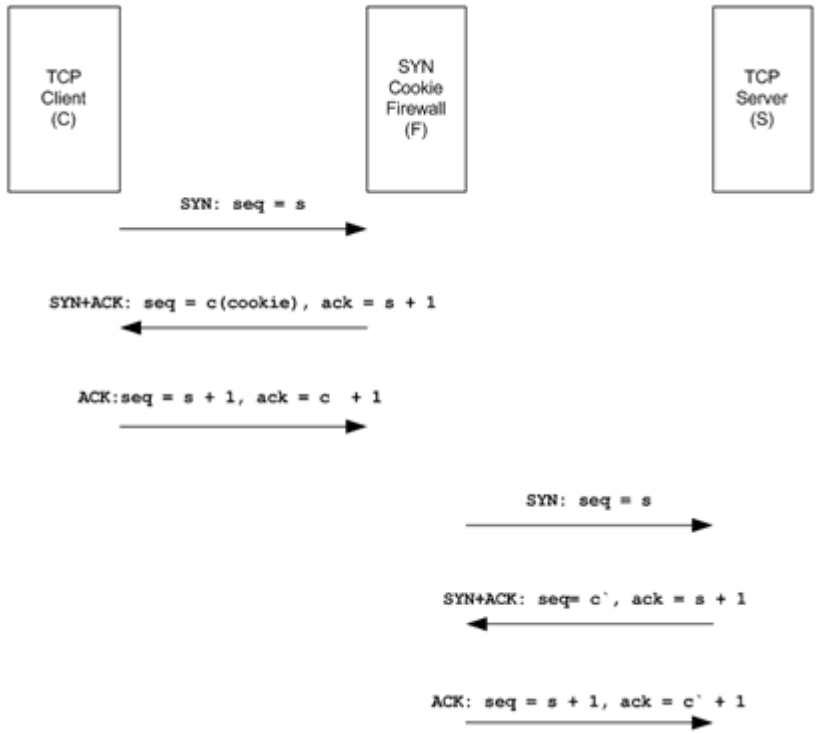
对于 SYN 攻击，SYN Cache 虽然不分配 TCB，但是为了判断后续对方发来的 ACK 报文中的 Sequence Number 的正确性，还是需要使用一些空间去保存己方生成的 Sequence Number 等信息，也造成了一些资源的浪费。Syn Cookie 技术则完全不使用任何存储资源，这种方法比较巧妙，它使用一种特殊的算法生成 Sequence Number，这种算法考虑到了对方的 IP、端口、己方 IP、端口的固定信息，以及对方无法知道而己方比较固定的一些信息，如 MSS、时间等,在收到对方 的 ACK 报文后,重新计算一遍,看其是否与对方回应报文中的(Sequence Number-1）相同，从而决定是否分配 TCB 资源。

3．使用 SYN Proxy 防火墙

SYN Cache 技术和 SYN Cookie 技术总的来说是一种主机保护技术，需要系统的 TCP/IP 协议栈的支持，而目前并非所有的操作系统支持这些技术。因此很多防火墙中都提供一种 SYN 代理的功能，其主要原理是对试图穿越的 SYN 请求进行验证后才放行，下图描述了这种过程：

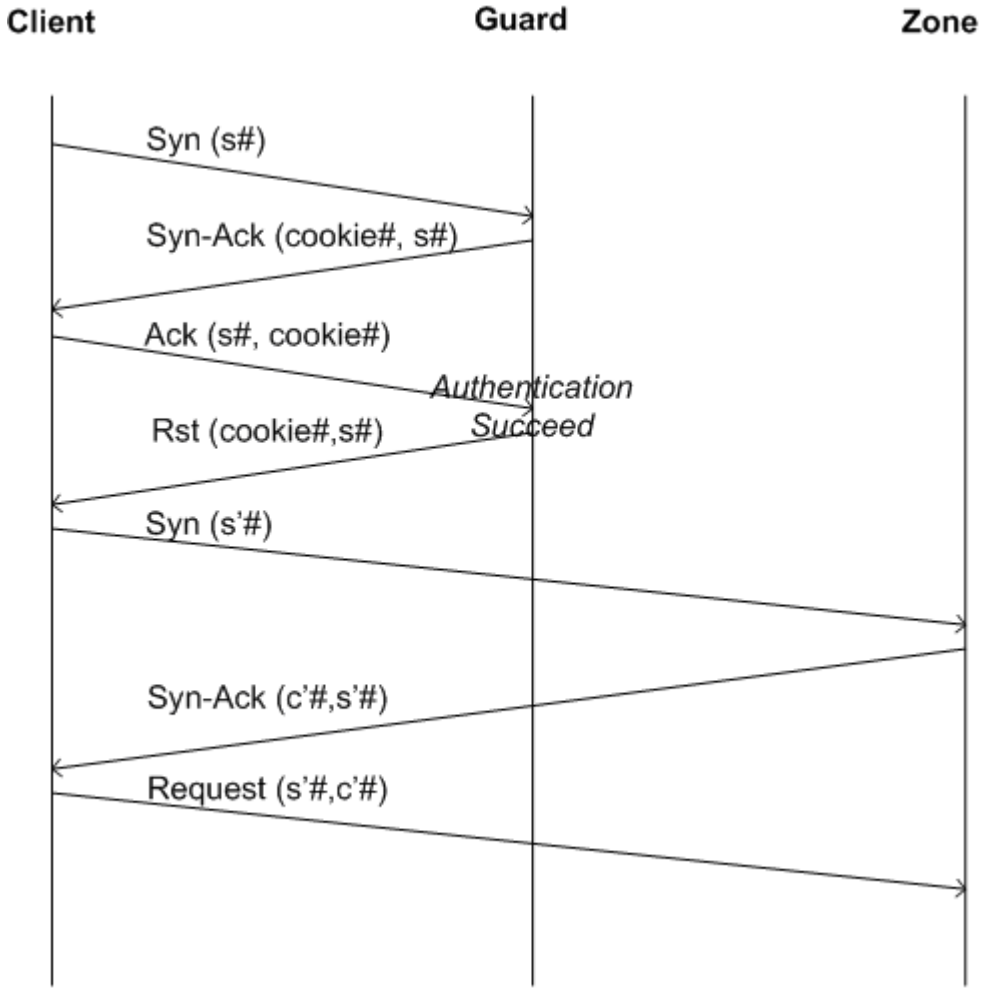


从上图（左图）中可以看出，防火墙在确认了连接的有效性后，才向内部的服务器（Listener）发起 SYN 请求，在右图中，所有的无效连接均无法到达内 部的服务器。而防火墙采用的验证连接有效性的方法则可以是 SYN Cookie 或 SYN Cache 等其他技术。采用这种方式进行防范需要注意的一点就是防火墙需要对整个有效连接的过程发生的数据包进行代理，如下图所示：



因为防火墙代替发出的 SYN ACK 包中使用的序列号为 c，而服务器真正的回应包中序列号为 c’，这其中有一个差值|c-c’|，在每个相关数据报文经过防火墙的时候进行序列号的修改。

TCP Safe Reset 技术：
这也是防火墙 SYN 代理的一种方式，其工作过程如下图所示：



这种方法在验证了连接之后立即发出一个 Safe Reset 命令包，从而使得 Client 重新进行连接，这时出现的 Syn 报文防火墙就直接放行。在这种方式中，防火墙就不需要对通过防火墙的数据报文进行 序列号的修改了。这需要客户端的 TCP 协议栈支持 RFC 793 中的相关约定，同时由于 Client 需要两次握手过程，连接建立的时间将有所延长。